THE CLAIMS

1. In a computer including hardware, a virtual machine monitor, and first and second operating system instances, a method comprising:

using the virtual machine monitor to expose the first operating system instance to a first hardware partition and prevent the first operating system instance from discovering a second hardware partition;

using the virtual machine monitor to expose the second operating system instance to the second hardware partition and prevent the second operating system instance from discovering the first hardware partition; and

using the virtual machine monitor to share at least some of the hardware among the first and second operating system instances.

2. The method of claim 1, wherein the first and second partitions include different portions of memory.

3. The method of claim 1, wherein the first and second partitions include different I/O devices.

4. The method of claim 1, wherein the shared hardware includes a CPU.

5. The method of claim 1, wherein the virtual machine monitor allows the first instance to have direct control over the first partition, and the second instance to have direct control over the second partition.

6. The method of claim 1, wherein the virtual machine monitor configures the hardware so accesses to addresses of interest trap to the

VMM; and wherein the first and second instances are booted on the virtual machine monitor after the hardware has been configured.

7. The method of claim 6, wherein the virtual machine monitor uses memory management to partition I/O devices.

8. The method of claim 7, wherein the VMM configures the hardware to trap to the VMM either when an access misses in a translation lookaside buffer, or when one of the operating system instances modifies its page table.

9. The method of claim 7, wherein the hardware is configured to treat physical addresses as virtual addresses, whereby the virtual machine monitor also uses memory management to trap accesses to physical addresses.

10. The method of claim 9, wherein the hardware includes a CPU, and wherein the virtual machine monitor configures the CPU to disable direct accessibility of the physical memory, whereby the VMM can trap I/O and physical memory accesses.

11. The method of claim 7, wherein using the memory management includes inspecting an address translation on a trap and modifying, accepting, or rejecting the translation.

12. The method of claim 7, wherein using the memory management includes inserting translations for I/O addresses into a translation lookaside buffer or page table.

13. The method of claim 12, wherein the virtual machine monitor grants unfettered access by an operating system instance to the range of physical memory covered by the translation entry in its translation lookaside buffer or page table.

14. The method of claim 6, wherein the traps occur during resource discovery of a booting operating system instance; and wherein the virtual machine monitor responds to a trap by misinforming the booting OS instance about the existence of hardware not in its partition.

15. The method of claim 1, wherein the virtual machine monitor modifies a hardware description table to expose and prevent discovery.

16. The method of claim 1, wherein the virtual machine monitor performs emulation to share hardware.

17. The method of claim 1, further comprising:

delivering interrupts to interrupt handlers maintained by the first instance when the first instance accesses the first partition; and

delivering interrupts directly to interrupt handlers maintained by the second instance when the second instance accesses the second partition.

18. The method of claim 1, wherein operation of the virtual machine monitor is transparent to the first and second operating system instances.

19. The method of claim 1, wherein the virtual machine monitor partitions I/O devices bus-wise.

20. In a computer including hardware, a virtual machine monitor running on the hardware, a method comprising:

booting a plurality of operating system instances on the virtual machine monitor;

using the virtual machine monitor to expose each of the booting operating system instances to its own partition and to prevent each of the operating system instances from discovering other hardware partitions; and

using the virtual machine monitor to share at least some of the hardware among the operating system instances;

wherein operation of the virtual machine monitor is transparent to the plurality of operating system instances.

21. The method of claim 20, wherein the virtual machine monitor can configure the hardware so accesses to addresses of interest trap to the VMM; and wherein the operating system instances are booted on the virtual machine monitor after the hardware has been configured.

22. The method of claim 20, wherein the hardware includes a CPU, and wherein the virtual machine monitor can configure the CPU to disable direct accessibility of the physical memory, whereby the VMM can trap I/O and physical memory accesses.

23. The method of claim 20, wherein the virtual machine monitor uses memory management to partition I/O devices, the memory management including inspecting an address translation on a trap and modifying, accepting, or rejecting the translation.

24. The method of claim 20, wherein the virtual machine monitor uses memory management to partition I/O devices, the memory management including inserting translations for I/O addresses into a translation lookaside buffer or page table.

25. The method of claim 24, wherein the virtual machine monitor grants unfettered access by an operating system instance to the range of physical memory covered by the translation entry in its translation lookaside buffer or page table.

26. The method of claim 20, wherein traps occur during resource discovery of a booting operating system instance; and wherein the virtual machine monitor responds to a trap by misinforming the booting OS instance about the existence of hardware not in its partition.

27. The method of claim 20, wherein the virtual machine monitor modifies a hardware description table to expose and prevent discovery.

28. The method of claim 20, wherein the virtual machine monitor partitions I/O devices bus-wise.

29. A processor and memory comprising:

means for running a virtual machine monitor; and

means for running first and second operating system instances on the virtual machine monitor;

the virtual machine monitor designed to expose a first operating system instance to a first hardware partition and prevent the first operating system instance from discovering a second hardware partition;

the virtual machine monitor designed to expose a second operating system instance to the second hardware partition and prevent the second operating system instance from discovering the first hardware partition;

the virtual machine monitor designed to allow at least some hardware sharing among the first and second operating system instances.

30. A computer for running first and second operating system instances, the computer comprising hardware including memory, the memory encoded with a virtual machine monitor for exposing the first operating system instance to a first partition of the hardware and preventing the first operating system instance from discovering a second partition of the hardware; exposing the second operating system instance to the second hardware partition and preventing the second operating system instance from discovering the first hardware partition; and sharing at least some of the hardware among the first and second operating system instances.

31. The computer of claim 30, wherein the virtual machine monitor can configure the hardware so accesses to addresses of interest trap to the VMM; whereby operating system instances can be booted on the virtual machine monitor after the hardware has been configured.

32. The computer of claim 30, wherein the hardware includes a CPU, and wherein the virtual machine monitor can configure the CPU to disable direct accessibility of the physical memory, whereby the VMM can trap I/O and physical memory accesses.

33. The computer of claim 30, wherein the virtual machine monitor can use memory management to partition I/O devices, the memory

management including inspecting an address translation on a trap and modifying, accepting, or rejecting the translation.

34. The computer of claim 30, wherein the virtual machine monitor can use memory management to partition I/O devices, the memory management including inserting translations for I/O addresses into a translation lookaside buffer or page table.

35. The computer of claim 30, wherein traps occur during resource discovery of a booting operating system instance; and wherein the virtual machine monitor can respond to a trap by misinforming the booting OS instance about the existence of hardware not in its partition.

36. The computer of claim 30, wherein the virtual machine monitor can modify a hardware description table to expose and prevent discovery.

37. The computer of claim 30, wherein the virtual machine monitor can partition I/O devices bus-wise.

38. An article for a computer, the article comprising computer memory encoded with a virtual machine monitor for exposing a first operating system instance to a first hardware partition and preventing the first operating system instance from discovering a second hardware partition; exposing a second operating system instance to the second hardware partition and preventing the second operating system instance from discovering the first hardware partition; and sharing at least some of the hardware among the first and second operating system instances.

39. The article of claim 38, wherein the virtual machine monitor can configure the hardware so accesses to addresses of interest trap to the VMM; whereby operating system instances can be booted on the virtual machine monitor after the hardware has been configured.

40. The article of claim 38, wherein the hardware includes a CPU, and wherein the virtual machine monitor can configure the CPU to disable direct accessibility of the physical memory, whereby the VMM can trap I/O and physical memory accesses.

41. The article of claim 38, wherein the virtual machine monitor can use memory management to partition I/O devices, the memory management including inspecting an address translation on a trap and modifying, accepting, or rejecting the translation.

42. The article of claim 38, wherein the virtual machine monitor can use memory management to partition I/O devices, the memory management

including inserting translations for I/O addresses into a translation lookaside buffer or page table.

43. The article of claim 38, wherein the traps occur during resource discovery of a booting operating system instance; and wherein the virtual machine monitor can respond to a trap by misinforming the booting OS instance about the existence of hardware not in its partition.

44. The article of claim 38, wherein the virtual machine monitor can modify a hardware description table to expose and prevent discovery.

45. The article of claim 38, wherein the virtual machine monitor can partition I/O devices bus-wise.